C2 - Python for Data Science

C-DAT-100

Web Scraping

Finding HTML tags in a haystack

1.0

Web Scraping

delivery method: py04 on Github
language: python

Web scraping means extracting data from websites in a automated way.

In 1998, Google was the first to do so; **Search engines are using web scraping to retreive HTML tags** from public website in order to rank them.

Web scraping is also used for collecting data **when no official API is available**.

Be carefull, web scraping may by illegal depending of the country and their legislation.

```
                </div>
            </div>
        </div>
    </div>
    <div class="wrap__2stroke">
        <div class="second">
            <div class="second__content">
                <div class="block">
                </div>
```

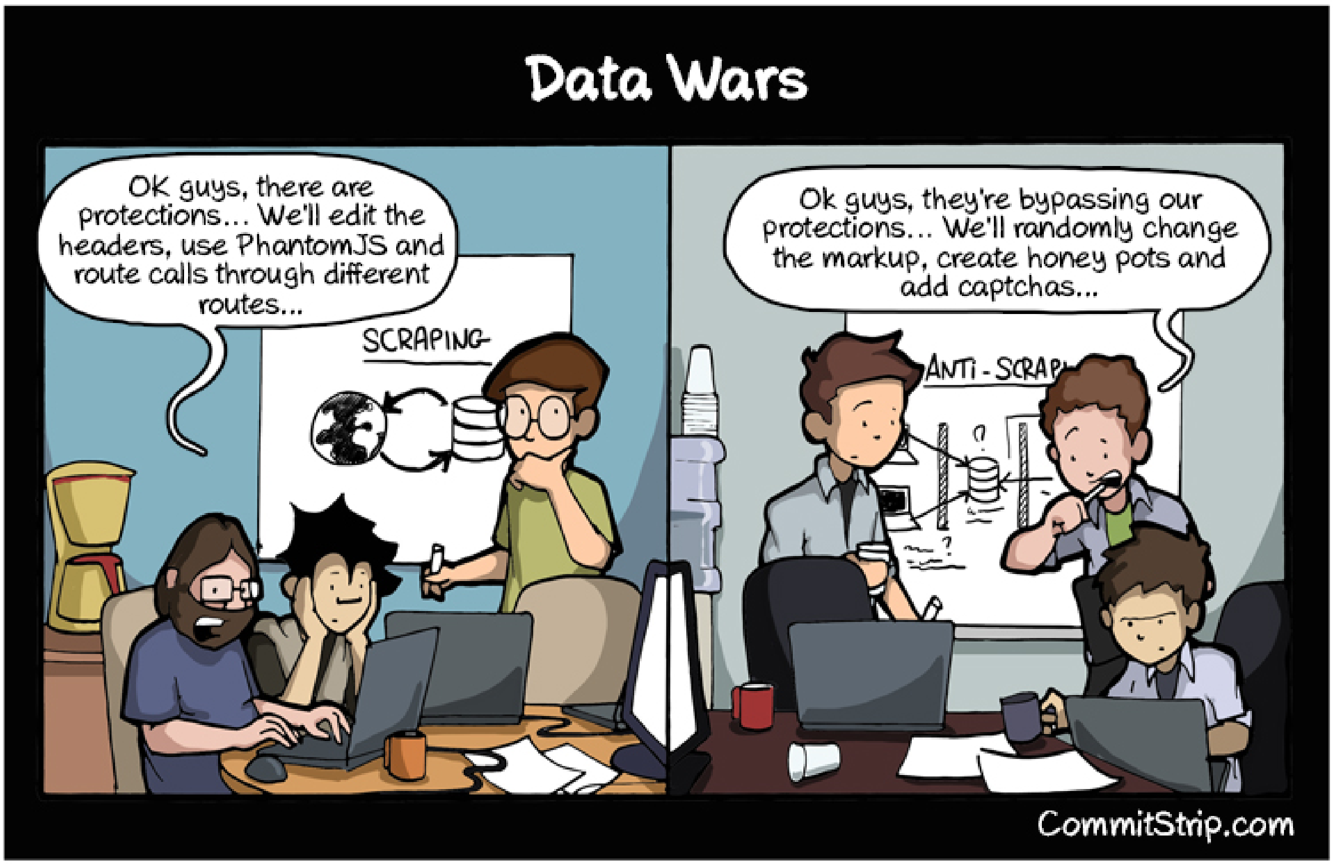
INTRO

Perform a GET request to **https://www.leboncoin.fr** to reteive the HTML homepage.

You must use an **user agent** to do so, as basic web security prevent HTTP requests from unknown web browser.

You can give this one a try :

Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.3945.74 Safari/537.36 Edg/79.0.309.43


Data Wars
OK guys, there are protections... We'll edit the headers, use PhantomJS and route calls through different routes...
SCRAPING
Ok guys, they're bypassing our protections... We'll randomly change the markup, create honey pots and add captchas...
ANTI-SCRAP
CommitStrip.com

Exercice

Create a function **get_ps5_prices()** that returns data from Playstation 5 game console sold on the website using **BeautifulSoup** python library.

Route : **/recherche?category=43&text=ps5**

You must retreive for each ads

- the **title**

- the **price** of the article (sellers may not have set a price for the article, put 0 instead)

- the **date** of when it was posted as **ISO8601 format**

- the **city**

- the **postal code**

Store all the data into a pandas dataframe.

Do only the first page (no need to go to page 2….3…..4)

Tips

The HTML you receive from your HTTP request is a basically a **snapshot** of the website as if the search was done from a web browser such as Firefox.

You should do this search from your web browser **at the same time** and use the **web inspector** to identify which HTML tags are relevant to get the data correctly.

Exercise 02

Rather than exporting the pandas dataframe into SQL database, export it as a file stored on your computer.

This is called **serialization**.

Export your data using **pickle**, name the file **ps5-dataframe.pickle**

Create a loop that call **get_ps5_prices** function every 5 minutes using **time.sleep(300)**

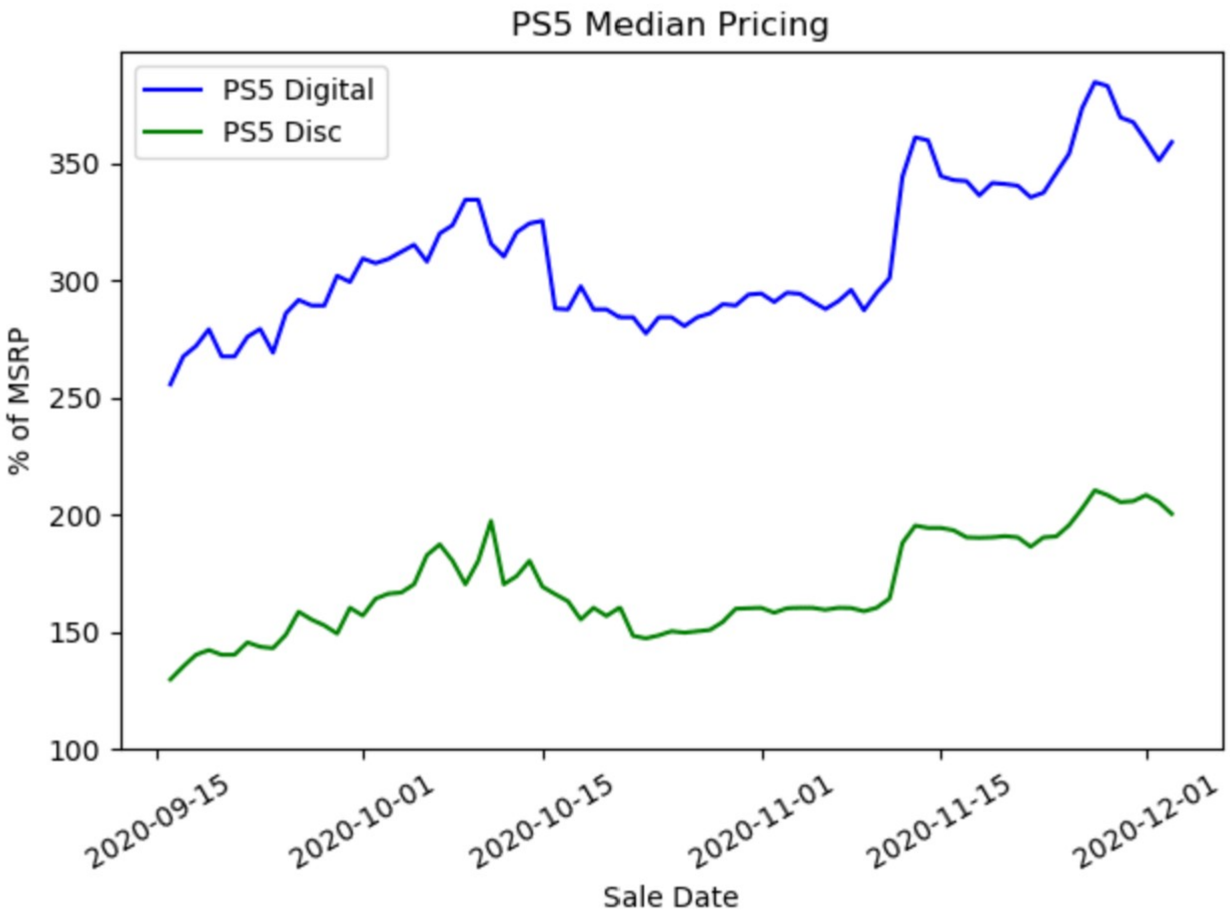
Only if a new article was published as compared to the previous iteration, **no duplicate data**, export it again.

Exercise 03

Create a new notebook call ex03

Create loop. Every 5 minutes, open `ps5-dataframe.pickle` in **read-only mode** in order to have your pandas dataframe back.

Using Seaborn, **line plot** the price.


PS5 Median Pricing
PS5 Digital
PS5 Disc
% of MSRP
Sale Date